UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/684,552	10/14/2003	Sophia Krasikov	YOR920030279US1	5614

48813 7590 07/22/2008
LAW OFFICE OF IDO TUCHMAN (YOR)
ECM #72212
PO Box 4668
New York, NY 10163-4668

EXAMINER	
WEI, ZHENG	

ART UNIT	PAPER NUMBER
2192	

NOTIFICATION DATE	DELIVERY MODE
07/22/2008	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

pair@tuchmanlaw.com
idotuchman@gmail.com

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/684,552
Filing Date: October 14, 2003
Appellant(s): KRASIKOV ET AL.

Ido Tuchman
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 05/16/2008 appealing from the Office action mailed 12/14/2007

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

5,987.251 Crockett 11-1999

WinZip, WinZip (R) Self-Extractor Personal Edition. Oct., 2002

FCC, "FCC Computer Security Notice", June, 2002

Heesch, "Doxygen Manual for version 1.2.10", 2001

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

- Claims 26-29 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and/or use the invention.

Claim 26: Applicant claims "at least one block of source code providing executable computer instructions" at page 16 lines 1-2 [emphasis added]. However, the "source code" itself cannot provide "executable computer instructions" without passing compilation or interpretation and generating executable code phases as being well known in the art. The originally filed specification at page 5, lines 1-3, discloses the source code is a high level programming language code which needs to be built to generate an executable unit. Although, the Applicants simply mention this aspect of the invention at page 4, lines 13-16 of the specification, they do not provide detail information about what this kind of source code is that can directly provide computer executable instruction. Therefore, it is not enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and/or use the invention. For the purpose of compact prosecution, the Examiner treats the "source code" as –high level programming language code and need to be compiled to provide executable computer instructions--.

Claims 27-29: Claims 27-29 are dependent claims of claim 27. Therefore they are also rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement.

- Claims 26-29 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claim 26: Claim 26 claims a program code that is embodied on computer-readable media and comprised a block of source code, a documentation string and indicia. However, said program code does not appear to produce/perform any functions so as to satisfy the practical application requirement. Therefore, it is directed to nonfunctional descriptive material. When nonfunctional descriptive material is recorded on some computer-readable medium, in a computer or on an electromagnetic carrier signal, it is not statutory since no requisite functionality is present to satisfy the practical application requirement. See M.P.E.P. 2106.01 (II).

Claims 27-29: Claims 27- 29 are dependent claims of claim 26. These claims all fail to remedy the 35 USC 101 nonstatutory problem of claim 26. Therefore, they are all not statutory. See M.P.E.P. 2106.01 (II).

- Claims 1-8 and 14-25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Crockett (Crockett et al., US 5,987,251) in view of WinZip (WinZip® Self-Extractor 2.2 product).

Claim 1:

Crockett discloses a method for creating a minimum executable unit, the method comprising:

- receiving at least one block of source code (see for example, Fig.5 step 52 "Pass JCL Instructors into Documentation Checker Tool" and related text);
- receiving at least one document element associated with the source code (see for example, step 56, "Pass documentation into Documentation Checker Tool" and related text);
- demarcating a target audience of the document element (see for example, Fig.6, element 64, 68, "Element: Job Name, Job Step Name and Proc Step Name" and related text at col.6, lines 40-64; also see col.5-6, example code by using tag e.g. "JOB NAME"… to identify/demarcate the desired document element)

- validating consistency between the source code and the document element (see for example, step 58, “Documentation Checker Tool Compares documentation dataset with job dataset” and related text);

But does not disclose following:

- building an executable unit, the executable unit being responsive to both the source code and the document element.

However, it is well known in the computer art that WinZip tool has a feature to generate the executable unit (self-extracting archive) (see for example, p.1, section “What is a Self-extracting Zip File?”) to include both the source code and the document element. Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to use WinZip’s function to generate the executable unit (self-extracting archive file). One would have been motivated to do so to automatically extracted and easily deploy files in archive as suggested by WinZip (see for example, p.1-2, section “What are the advantages to Self-extracting Zip Files?”)

Claim 2:

Crockett and WinZip disclose the method of claim 1, Crockett further discloses said method comprising receiving at least one external document for inclusion in the executable unit (see for example, Fig.5, step 48, “Pass documentation format information to Documentation Checker Tool” and related text).

Claim 3:

Crockett and WinZip disclose the method of claim 1, Crockett further discloses, wherein the document element includes tag-based syntax , the tag-based syntax including the target audience of the document element (see for example, Fig.5, step 50, “Documentation Checker Tool builds list of tags and text” and also see col.7, lines 40-45,”The documentation checker tool uses the documentation format information to locate elements within the documentation …”; also see Fig.6, element 64, 68, “Element: Job Name, Job Step Name and Proc Step Name” and related text at col.6, lines 40-64).

Claim 5:

Crockett and WinZip disclose the method of claim 1, but neither of them discloses said method further comprising providing a text editor configured to create the document element with tag-based syntax to demarcate information, the tag-based syntax including the target audience of the document element (see for example, Fig.6, element 64, 68, “Element: Job Name, Job Step Name and Proc Step Name” and related text at col.6, lines 40-64). However, it is well known in the computer art that such text based document element (text file) can be created by any kind of text based editor tool, e.g., those default edit tools: Vi. Emacs, Notepad or WordPad according to different OS platforms. Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to create document element by using any kinds of text based editing tools.

Claim 6:

Crockett and WinZip disclose the method of claim 1. It is also well known in the computer that the executable unit (self-extracting file) can be extracted by itself and further extracting the document element from the unzipped files in such a way as to be readable as documentation. Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to get the documents element by running executable unit.

Claim 7:

Crockett and WinZip disclose the method of claim 1, it is also well known in the computer art that text based file can be created, edited by text based editing tools. Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to aggregate the Crockett's document element with other document elements to create a new document file.

Claim 8:

Crockett and WinZip disclose the method of claim 7. It is also well known in the computer art that the executable unit (self-extracting file) can be extracted by itself and further extracting the document element from the unzipped files. Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to aggregate other document elements which are extracted from other executable units to create new document file more efficient.

Claims 14-16:

Claims 14-16 are system version for performing the claimed method as in claims 1 and 2 addressed above, wherein all claimed limitation functions except the “human audience”, have been addressed and/or set forth above and certainly a computer system would need to run and/or practice such function steps disclosed by Crockett (see for example, col.19, lines 19-45, also see Fig.1 and related text) and WinZip. As in claim 1, Crockett discloses the limitation about “target audience”, but does not explicitly disclose such audience is human audience. However, Crockett also discloses using element tags to decide what elements need to be documented and where the text contents belong to (see for example, Fig.6, and related text at col.6, lines 40-64, “The documentation checker tool 20 must next process the JCL instructions 24 in a job 22 in order to identify what elements need to be documented in the documentation”; also see col.5-6, the example of an appropriate documentation format). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to use the element tags to identify the different target human audience, the checker in Crockett can decide where the document element is targeted to according the predefined the template as disclosed (col.5-6, the example of an appropriate documentation format). Thus, they also would have been obvious.

Claims 17-19 and 21-24:

Claims 17-19 and 21-24 are computer program products version of the claimed method, wherein all claimed limitation functions have been addressed in claims 1-3, 6-9 and 14-16 above respectively. It is well known in the computer art that such method steps can be implemented as computer program and can be practiced and /or stored on a computer operable media. Thus, they also would have been obvious in view of Crockett and WinZip's teachings.

- Claims 9 and 25-29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Crockett (Crockett et al., US 5,987,251) in view of WinZip (WinZip® Self-Extractor) in further view of FCC (FCC Computer Security Notice)

Claim 9:

Crockett and WinZip disclose the method of claim 1, WinZip further discloses a feature of using encryption (a feature of ZIP 2.0 standard) (see for example, p.2, section "Summary of Key Features", "100% compatible with the ZIP 2.0 standard") and FCC also discloses the same feature about encryption of WinZip (see for example, p.2, line 2, “WinZip uses the industry standard ZIP 2.0 encryption format”). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to use the encryption feature provide by WinZip to encrypt documentation file in Crockett to protect sensitive documents as suggested by WinZip (see for example, p.2, lines 2-3, “WinZip uses the industry standard ZIP 2.0 encryption format. Password protecting files in a Zip file provides a measure of protection against casual users viewing the files.”)

Claim 25:

Claim 25 is computer program products version of the claimed method, wherein all claimed limitation functions have been addressed in claims 1-9 above respectively. It is well known in the computer art that such method steps can be implemented as computer program and can be practiced and /or stored on a computer operable media. Thus, they also would have been obvious in view of Crockett, WinZip and FCC’s teachings.

Claim **26-29**:

Claims 26-29 are computer program products version of the claimed method, wherein all claimed limitation functions have been addressed in claims 1-9 above respectively. It is well known in the computer art that such method steps can be implemented as computer program and can be practiced and /or stored on a computer operable media. Thus, they also would have been obvious in view of Crockett, WinZip and FCC's teachings.

- Claims 30 and 31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Crockett (Crockett et al., US 5,987,251) in view of WinZip (WinZip® Self-Extractor) in further view of Heesch (Dimitri van Heesch, Doxygen Manual for version 1.2.10)

Claim 30:

Crockett and WinZip disclose the method of claim 1, but neither of them explicitly discloses wherein the document element is embedded in the source code. However, Heesch in the same analogous art of documenting program source code by embedded documentation information in the source code and auto generate documentation by running Goxygen application (see for example, p.12, section 2.2, "step 2: Running doxygen" and related text and section 2.3, "step 3: Documenting the sources"). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to embed all documentation information including all the documentation format information in the source code by using Doxygen's special documentation blocks. One would have been motivated to do so to embed the document in the source code to direct generated on-line documentation and off-line reference manual from source files which makes it much easier to keep the documentation consistent with the source code as suggested by Heesch (see for example, p.1, lines 13-17, "The documentation is extracted directly form the sources, which makes it much easier to keep the documentation consistent with the source code.")

Claim 31:

Claim 31 is computer program product version of the claimed method, wherein all claimed limitation functions have been addressed in claims 1, 17 and 30 above respectively. It is well known in the computer art that such method steps can be implemented as computer program and can be practiced and /or stored on a computer operable media. Thus, they also would have been obvious in view of Crockett, WinZip and Heesch's teachings.

(10) Response to Argument

A. Claim 26 (Brief pages 5-6)

The Appellant argues that the rejection of claim 26 under 35 U.S.C. § 112, first paragraph, is in error. Because "it is well known to those skilled in the art that source code may be compiled or interpreted", and thus "In interpreter languages, an executable program can be source doe in the proper format".

In response, it should be noted that the "interpreter languages" itself is not computer executable instruction which directs the computer hardware to perform certain functions/steps [emphasis added]. It is the interpreter provides the computer executable instructions while it reads the proper format of interpreter language instructions (once at a time) and generates computer executable instructions (emphasis added). Although the source code could be the interpreter language, it does not directly provide computer executable instruction (emphasis added). As the Appellant argued at second paragraph of page 6, “In interpreter languages, an executable program can be source code in the proper format.”, that is to say that with the proper interpreter language source code, under an interpreter, can be used as executable source code and thus the source code itself does not directly provide executable instructions (emphasis added). Because it, the proper interpreter language, has to be interpreted by the interpreter prior to generate/provide executable instructions. Moreover, the specification does not provide any information about source code providing (directly) computer executable instructions. The Examiner can only be able to find, for example, Fig.2 illustrates as item 204 “builder” to combine the source code and document elements to generate executable code (item 206) (indirectly - emphasis added), and Fig.3 with an example of source code with embedded document element. However, neither of them shows anything about the "computer executable instructions” since there is no activities/acts that the executable instructions require to be performed. Therefore, it is not enable to one skilled in the art to which it pertains or with which it is most nearly connected, to make and/or use the invention.

B. Claim 26 (Brief pages 6-7)

The Appellant submits that “claim 26 is directed to functional material recorded on computer-readable medium, and is statutory subject matter under 35 U.S.C. § 101”.

In response, it should be noted that "the program code" in claim 26 merely comprises one block of source code providing executable computer instructions(110) and one document string (112) as showed in Fig 2, that have to use “Builder” (204) to build/generate an executable unit (Item 206, “Minimum Executable Unit”). First of all, the program code with the block of source code cited in the claim is not an executable code by itself (directly) as addressed above in section A. Secondly, even arguably, the source code providing (indirectly) the computer executable instructions, one may ask that what steps/activities/acts are the executable instructions require to be performed in the claim? The answer is “NONE”. There are no “acts” actually being performed by the executable instructions, thus, there is no functionalities to be realized. Therefore, the program code is just computer program code listings *per se* without program’s functionality to be realized. Moreover, there is no functional interrelationships between the source code and the document string as recited in the claim, thus, even when the “program code embodied on computer-readable media”, there is no program functionality to be realized and/or as they are not “acts” actually being performed by such a computer. Therefore, the so called and claimed “program code” including “one block of source code” and “one documentation string” is directed to nonfunctional descriptive material and thus claim 26 is not statutory. *See* M.P.E.P. 2106.01 (II).

C. Claims 1-8 and 14-25 (Brief pages 8-11)

The Appellant argues that the Examiner’s interpretation of Crockett is in error and the figure 6 of Crockett is not demarcating a target audience of the document element. However, the Examiner respectfully disagrees. Figure 6 of Crockett shows documentation formatting information (item 62) including “Job Name”, “Job Step Name” and "Proc Step Name" which are used to identify each of the desired elements (emphasis added) (see for example, col.6, lines 42-44, states "The element or tag indicates an element to be included in the documentation..."; also see col.6, lines 45-49, "Hence, the job name element has the associated text "Job Name:". Accordingly, each of the desired elements may be identified within the documentation format information 62 to realize documentation like the specified above." (Emphasis added). Furthermore, as Crockett disclosed an example of an appropriate document format (col.5, line 10-col.6, line 34) clearly indicates the documentation format information, e.g., "JOB NAME", "JOB NARRATIVE" which will reference by the documentation checker tools (item 20) to demarcating/identifying the target audience of the document element. Therefore, Crockett does disclose the feature by using element/tag, e.g., "Job Name" to demarcate/indicate the desired element (target audience of the document element).

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Zheng Wei/

Examiner, Art Unit 2192 Thursday, July 10, 2008

Conferees:

/Tuan Q. Dam/
Tuan Q. Dam
Supervisory Patent Examiner, Art Unit 2192

/Lewis A. Bullock, Jr./
Supervisory Patent Examiner, Art Unit 2193